Diabetes MythBuster: Automated Fact-Checking for Arabic Videos About Diabetes

Group Members

Mohammed Khalid Alyobi.

Ahmed Saleh Ahmed Akhtarulzaman.

Ahmad Meshaal Aljedaani.

Supervisor

Mr. Elias Al Ghazal.

Abstract

The rapid spread of diabetes-related misinformation in Arabic short-form videos poses significant public health risks, yet existing automated fact-checking systems are predominantly designed for English content. This project presents *Diabetes MythBuster*, a modular pipeline that integrates Speech-to-Text (STT), claim extraction, and evidence-based fact verification to automate fact-checking of Arabic health videos. The system employs Whisper-Large for accurate transcription, Gemini Flash 1.5 for claim extraction and verdict classification, and a hybrid fact-checking strategy that combines domain-filtered web search with Retrieval-Augmented Generation (RAG) via ChromaDB. Evaluation demonstrated that Whisper-Large achieved superior transcription quality (Jaccard = 0.805, recall = 0.884, precision = 0.904) compared to alternatives. For claim extraction, the final pipeline improved precision and recall balance (F1 = 0.78) over the baseline (F1 = 0.71), while verdict classification achieved near-expert performance (Macro F1 = 0.95 vs. baseline 0.60). These findings highlight the feasibility of deploying evidence-grounded LLM pipelines to combat medical misinformation in Arabic digital spaces. Future work will extend the system to additional health domains, enhance robustness to noisy inputs, and integrate multi-modal analysis for broader misinformation detection. This is a reference to the repo we developed:

https://github.com/asaazamaan/factChecker

1. Introduction

Social media platforms are widely used for health information sharing, but they also serve as hotbeds for misinformation. In the Arabic-speaking world, diabetes-related misinformation spreads particularly fast, often misleading individuals with false or unproven claims about treatments or cures. This misinformation poses serious health risks, leading people to delay or avoid legitimate medical care.

This project addresses this challenge by building an automated system that can detect and verify health-related claims found in short online Arabic videos. The solution aims to help users distinguish factual content from misleading narratives using trusted medical sources.

2. Related Work

To establish a strong foundation for our system design, we explored a wide range of resources from the Automated-Fact-Checking-Resources repository (Contributers, n.d.), which has many works across claim detection, verification, and automated fact-checking systems. From this repository, we focused particularly on papers relevant to multimedia content and health misinformation. Notable examples include research employing neural architectures such as BERT, RoBERTa, and LSTM variants for identifying check-worthy claims in political and social contexts, approaches that could be adapted to health-related content in Arabic. While our project did not directly implement these specific architectures, the methodologies and findings from these works provided valuable insights that informed our design decisions. In parallel, we also reviewed documentation for various Large Language Models (LLMs) (Google, Gemini API Documentation , n.d.) to better understand and utilize their APIs effectively within our system.

In addition to these curated works, we reviewed tools related to STT tasks. The Whisper paper from OpenAI (AI, n.d.) was the main highlight, which details a robust multilingual Speech-to-Text (STT) system. In addition to other tools that provide STT services (e.g., GPT-4o mini).

Finally, informal consultations with researchers and students who had built similar pipelines provided valuable practical insight. The combination of theoretical insights and empirical findings helped shape our modular pipeline and informed key design decisions.

3. Approach

The initial phase of the project focused on establishing a functional baseline to assess the overall feasibility of the proposed approach, without prioritizing accuracy at this stage. A preliminary pipeline was developed, incorporating *yt-dlp* (yt-dlp, n.d.) for audio extraction, the Whisper Medium model for the Speech-to-Text (STT) task, and

the Gemini Flash 1.5 API for both claim extraction and preliminary judgment. This proof-of-concept confirmed the operational viability of the workflow, after which development progressed toward building a more detailed and robust system in incremental stages.

The system follows a modular fact-checking pipeline that consists of many steps to achieve the final goal of evaluating the claims:

1. **Video Handling**: The user is asked to paste the video link into the website MythBuster platforms' search bar. Then yt-dlp is used to download an audio file (MP3) from videos (e.g., YouTube). Audio segments are capped at 2 minutes for focused analysis; if the video is originally longer than 2 minutes, it will not be handled.

2. **Transcription**: Whisper-Large Through OpenAI API endpoint is used for an accurate performance in the STT task. Model inference is containerized using Docker with persistent volumes for model caching, if the same video is queried again, it will return the already stored result in the database instead of rerunning the whole process.

3. **Claim Extraction**: In this step, we use the Gemini Flash 1.5 API with an engineered prompt to extract factual claims from the already generated transcripts. The extraction is forced to be in a JSON format for easy handling in the code.

4. **Fact-Checking**: Integration of trusted sources is achieved through real-time web search using the Serper.dev search engine (Google, Serper, n.d.). The approach involves retrieving information from reputable sources on the internet (e.g., MOH) by applying domain-based URL filters. In parallel, Retrieval-Augmented Generation (RAG) is employed to retrieve relevant information from a curated research document collection (University, n.d.). For this, a ChromaDB (Chroma, n.d.) vector database is used to store and index the dataset, with embeddings generated using the all-MiniLM-L6-v2 model (SBERT.net, n.d.). Since the available resources are in English, the Gemini API is utilized to translate Arabic claims into English prior to similarity matching. The most relevant document chunks are then retrieved and appended as "context" to the Gemini prompt for fact evaluation. The system subsequently presents the user with each claim's classification as TRUE, FALSE, PARTIALLY_TRUE, or INSUFFICIENT_INFO, accompanied by Arabic medical reasoning and confidence scores generated by the LLM for each classification.

5. **Frontend and Backend**: FastAPI serves as the backend for processing video URLs and handling STT and claim extraction. A React-based frontend provides a user interface to upload video links and view fact-checking results. PostgreSQL is used for caching and storing outputs to avoid reprocessing the same inputs. And all are containerized using Docker to ensure consistency in the development requirements.

4. Experiments and Results

In order to choose the best-performing component in each step of the pipeline, we conducted a benchmarking experiment using multiple videos with multiple models for both test transcription and claim extraction. Here is a snapshot from the Excel file:



STT Results:

To evaluate the transcription quality of the Speech-to-Text (STT) component, we benchmarked four models—**Whisper-Medium**, **Whisper-Large**, **GPT-4o-mini**, and **GPT-4o**—across Arabic health-related videos. Ground truth transcriptions were used as the reference, and performance was assessed using three complementary metrics:

1. **unique_jaccard_similarity** – A *set-based* vocabulary similarity metric (range: 0–1) that compares only the unique words present in both the Ground Truth (GT) and the transcription, ignoring duplicates. This metric answers the question: *"Do the GT and transcription share the same vocabulary?"* It does not consider word order or frequency. Higher scores indicate that the transcription captured the correct set of words at least once.
2. **token_overlap_ratio_gt** – A *recall-oriented* metric that counts matching words (including duplicates) between GT and transcription, divided by the total GT word count. This shows what proportion of the original GT words were successfully captured.
3. **token_coverage_ratio_trans** – A *precision-oriented* metric that counts matching words (including duplicates) but divides by the total transcription length. This measures the proportion of transcription words that are correct.

The results averaged were as follows:

- **Whisper-Large**: avg. Jaccard = 0.805, avg. overlap recall = 0.884, avg. coverage precision = 0.904
- **GPT-4o**: avg. Jaccard = 0.684, avg. recall = 0.807, avg. precision = 0.815
- **GPT-4o-mini**: avg. Jaccard = 0.661, avg. recall = 0.770, avg. precision = 0.811
- **Whisper-Medium**: avg. Jaccard = 0.405, avg. recall = 0.458, avg. precision = 0.666

These results show a clear performance hierarchy. Whisper-Large consistently outperformed all other models across all metrics, indicating both strong vocabulary matching and a balanced precision–recall profile. GPT-4o and GPT-4o-mini delivered competitive results, with GPT-4o slightly ahead in recall and precision, but still below Whisper-Large in every metric. Whisper-Medium showed substantial weaknesses in both vocabulary coverage and token accuracy, making it unsuitable for high-accuracy Arabic transcription tasks.

Based on these empirical findings, Whisper-Large was selected as the STT model for the final system, as it provided the best trade-off between capturing the correct vocabulary, maximizing the proportion of ground truth words, and ensuring that most words in the transcription were correct.

Claim Extraction and Verification Results: We used multiple engineered commands, trying and figuring out which one is best. The last approach was to use two commands:

prompt = f"""

You are an expert medical claim extractor for an "AI MythBuster" project. Your task is to analyze the provided video transcript to identify and extract factual claims ONLY.

You must follow these rules strictly:

1 .The transcript may contain spelling errors, grammar mistakes, and conversational fillers. Ignore these errors and focus on understanding the intended meaning of the text.

2 .Read the transcript carefully to determine if the primary topic and claims are related to the medical field of "Diabetes" or diabetic care.

3 .If the transcript's claims are NOT about Diabetes, your output MUST be this exact JSON object:

"}} domain_is_diabetes": false, "claims{{[] :"

4 .If the transcript's claims ARE about Diabetes, identify the main claims.

- *Prioritize extracting any claims that are likely to be medically misleading, exaggerated, or incorrect. If no such claims are found, then extract up to three significant factual claims. All claims must be written in correct Arabic, clearly and objectively, while preserving the speaker's original intended meaning without softening or interpreting it.*

- *Focus on the single most significant claim if possible.*

- *If a single main claim cannot be identified, extract up to a maximum of three distinct, verifiable claims.*

- *Each claim should be a concise, objective statement written in Arabic language in the `ar` field, and you must also provide a precise English translation for that claim in the `en` field. Do not put Arabic text in `en` or English text in `ar.`*

5 .Your output MUST be this exact JSON format, with the boolean value and claims array filled in based on the rules above. Do not include any other text, explanations, or conversational language. Return ONLY JSON. Do not wrap the JSON in code fences.

}}

" domain_is_diabetes": true,

" claims] :"

"}} ar": "Arabic claim text 1", "en": "English translation 1,{{"

"}} ar": "Arabic claim text 2", "en": "English translation 2,{{"

"}} ar": "Arabic claim text 3", "en": "English translation 3{{"

[

{{

Transcript to analyze:

"{transcript_text}"

"""

prompt = f"""

You are an expert medical fact-checker for diabetes claims. Evaluate each claim using the provided CONTEXT_SNIPPETS. These snippets may include medical literature (RAG) and trusted web evidence.

IMPORTANT: Use the relevant medical knowledge provided below to inform your fact-checking. If the provided documents contain information that supports or contradicts the claim, reference this in your reasoning. If context is insufficient, use established medical knowledge.

Verdict definitions:

-TRUE: Medically accurate according to the evidence

-FALSE: Medically incorrect or contradicted by the evidence

-PARTIALLY_TRUE: Contains some truth but is incomplete or misleading based on the evidence

-INSUFFICIENT_INFO: Cannot determine accuracy from the provided evidence

Return ONLY a JSON array of objects in the same order as input, with this exact schema:

}

" claim": "Arabic claim text,"

" verdict": "TRUE|FALSE|PARTIALLY_TRUE|INSUFFICIENT_INFO,"

" reasoning": "Brief medical explanation in Arabic,"

" medical_category": "treatment|prevention|symptoms|causes|diet|lifestyle"

{

Do not include any other text. Do not wrap the JSON in code fences.

You will now receive a JSON array named CLAIMS.

For each item, use the Arabic claim text (claim_ar) as the "claim" value, and the provided "context_snippets" to inform your judgement.

"""

These commands were tested with multiple Large Language Models (LLMs), including **Gemini 1.5 Flash, GPT-4o, GPT-4, and GPT-3.5**. The evaluation was conducted against ground truth labels generated by GPT-5 (LLM-as-Judge), with a focus on how accurately each model's outputs aligned with the original video content.

The results demonstrated that **Gemini 1.5 Flash** consistently achieved the best overall performance across both **claim extraction** and **verdict classification** tasks. In claim extraction, it produced precise and contextually relevant claims in Arabic, with significantly fewer hallucinations compared to GPT-4o or GPT-4. In verdict classification, the final pipeline (Gemini 1.5 Flash combined with Retrieval-Augmented Generation via ChromaDB and Serper.dev) reached an average **Macro F1 score of ~0.95**, far surpassing the base pipeline and other tested models.

GPT-4o and GPT-4 performed competitively, but they were slightly less reliable in maintaining consistent claim boundaries and occasionally produced verbose or overly cautious verdicts. **GPT-3.5**, however, performed poorly: in four separate cases, it incorrectly returned the response *"this video is not diabetes related"* even when the videos were explicitly about diabetes, highlighting its unsuitability for this domain.

From a quantitative perspective, the **claim extraction stage** showed that the base pipeline achieved a recall-oriented performance (Precision ≈ 0.65, Recall ≈ 0.78, F1 ≈ 0.71), capturing more claims but introducing significant noise. In contrast, the final pipeline balanced precision and recall more effectively (Precision ≈ 0.81, Recall ≈ 0.76, F1 ≈ 0.78), yielding **cleaner and more trustworthy claim boundaries** that directly improved downstream processing.

For the **verdict classification stage**, differences were even more pronounced. The base pipeline achieved only moderate results (Weighted Accuracy ≈ 0.70, Macro F1 ≈ 0.60), with frequent errors such as misclassifying `TRUE` claims as `PARTIALLY_TRUE` or contradicting them as `FALSE`. The final pipeline, however, reached near-expert performance (Weighted Accuracy ≈ 0.97, Macro F1 ≈ 0.95), with only rare mismatches between similar labels (`PARTIALLY_TRUE` vs `INSUFFICIENT_INFO`). This demonstrates

that retrieval grounding and evidence integration were **decisive in transforming raw LLM outputs into reliable fact-checking outcomes**.

System performance was optimized through a combination of caching, Docker-based containerization, and asynchronous processing. For a video processed for the first time, the full pipeline (transcription, claim extraction, and verdict classification) takes approximately **one minute**. However, if the video has been processed before, results are retrieved directly from the database, reducing response time to **under one second**.

5. Discussion

Our experiments demonstrate the viability of integrating **Large Language Models (LLMs)** with accurate **Speech-to-Text (STT)** and **Retrieval-Augmented Generation (RAG)** for fact-checking Arabic health-related videos. The modular pipeline — comprising video ingestion, STT, claim extraction, and fact verification — successfully processed short-form content and generated structured, evidence-grounded fact assessments in Arabic.

STT Performance. The benchmarking across four transcription models revealed that **Whisper-Large** consistently delivered the highest performance across all metrics: average unique Jaccard similarity of **0.805** (strong vocabulary match), token overlap ratio (**recall**) of **0.884**, and token coverage ratio (**precision**) of **0.904**. These values indicate that Whisper-Large not only captured most of the vocabulary present in the ground truth but also ensured that the majority of words it transcribed were correct. In contrast, **GPT-4o** and **GPT-4o-mini** showed competitive but consistently lower scores, while **Whisper-Medium** lagged significantly. This justified adopting Whisper-Large as the backbone for transcription in the final system.

RAG and Trusted Source Retrieval. Fact-checking relied on a hybrid approach combining **real-time web search** (via Serper.dev, with domain-based filters for trusted sources such as the Ministry of Health) and retrieval from a curated **research corpus** stored in ChromaDB. The corpus was embedded using **all-MiniLM-L6-v2**, while Arabic claims were translated to English via Gemini before similarity matching. The most relevant document chunks were then appended as context for Gemini during claim evaluation. This strategy improved coverage, enabling the system to verify claims that were not addressed by official health portals but documented in academic or clinical literature.

Pipeline Evaluation. A two-stage evaluation framework — covering both **claim extraction** and **verdict classification** — provided further insights. For claim extraction, the **base pipeline** achieved a recall-oriented performance (Precision ≈ 0.65, Recall ≈ 0.78, F1 ≈ 0.71), while the **final pipeline** delivered more balanced outputs (Precision ≈ 0.81, Recall ≈ 0.76, F1 ≈ 0.78), producing cleaner and more reliable claim boundaries. For verdict classification, the improvements were even more pronounced: the base pipeline reached only moderate alignment with ground truth (Macro F1 ≈ 0.60, Weighted Accuracy ≈ 0.70), whereas the final pipeline — with retrieval grounding and

evidence integration — achieved near-expert performance (Macro F1 ≈ 0.95, Weighted Accuracy ≈ 0.97).

Challenges and Limitations. Despite these promising results, several limitations remain. Poor audio quality, overlapping speech, or multiple speakers still reduced STT accuracy, sometimes causing cascading errors in claim extraction and fact verification. LLMs occasionally hallucinated — generating unsupported claims or verbose justifications — though constraint-driven prompting reduced the frequency of such errors. Additionally, emotionally charged or ambiguous claims (e.g., anecdotal "miracle cures") required nuanced reasoning that the system only partially addressed.

Future Directions. Addressing these limitations will require improvements such as: (1) incorporating **speaker diarization** and **noise reduction** before STT, (2) implementing **source reliability scoring** within RAG to better rank retrieved evidence, (3) expanding training and evaluation datasets to cover **diverse dialects and accents**, and (4) refining **LLM prompting and fine-tuning** to enforce more consistent, evidence-based reasoning.

Overall, the results confirm that **accurate STT, reliable retrieval, and controlled LLM outputs** are essential for building robust automated health fact-checking systems in Arabic. The shift from a recall-heavy baseline to an evidence-grounded final pipeline demonstrates how retrieval augmentation transforms raw LLM predictions into trustworthy, medically relevant outputs.

6. Conclusion

This project developed a **foundational pipeline for Arabic-language video fact-checking** with a focus on diabetes-related content. By integrating **Whisper-Large** for Speech-to-Text transcription, **Gemini Flash 1.5** for claim extraction and verdict classification, and a hybrid **fact verification strategy** combining real-time web search with **Retrieval-Augmented Generation (RAG)** via ChromaDB, the system demonstrated the feasibility of **automating health misinformation detection in Arabic short-form videos**. Evaluation across both claim extraction and verdict classification confirmed that retrieval grounding significantly improves precision and factual reliability compared to baseline LLM-only approaches.

Looking forward, several directions can further strengthen the pipeline:

- **Refinement of evaluation frameworks** — incorporating larger annotated datasets, more granular error analyses, and cross-model benchmarking.
- **Domain expansion** — extending beyond diabetes to other medical and public health topics.
- **Enhanced STT robustness** — improving handling of low-quality audio, dialectal variation, and multi-speaker overlaps.
- **Multi-modal claim detection** — integrating analysis of both video text and visual content alongside transcripts.
- **System optimization** — reducing latency, improving scalability, and developing an accessible application interface for real-world deployment.

Overall, the findings highlight that **evidence-grounded LLM pipelines can transition from proof-of-concept to practical health fact-checking tools**, supporting the fight against medical misinformation in Arabic digital spaces.

References

OpenAI, O. (n.d.). *Whisper*. Retrieved from https://github.com/openai/whisper

Chroma. (n.d.). *ChromDB Docs*. Retrieved from https://docs.trychroma.com/docs/overview/introduction

Contributers, G. (n.d.). *Automated-Fact-Checking-Resources*. Retrieved from Github: https://github.com/Cartus/Automated-Fact-Checking-Resources

Google. (n.d.). *Gemini API Documentation* . Retrieved from https://ai.google.dev/gemini-api/docs

Google. (n.d.). *Serper*. Retrieved from https://serper.dev/

SBERT.net. (n.d.). *Pretrained Models*. Retrieved from https://www.sbert.net/docs/sentence_transformer/pretrained_models.html

University, C. (n.d.). *arXiv*. Retrieved from arXiv: https://arxiv.org/

yt-dlp. (n.d.). Retrieved from https://github.com/yt-dlp/yt-dlp